C L A I M S

Sub a7

1. Method for controlling access to protected contents on a server, the method requiring the following components to be present:

a) a server

b) a client

c) a reader for a mobile security module

d) a security module having at least one protected area for storing a key

e) a data line for communications between client and server

characterized by the following steps:

aa) sending to the server of a request to call up protected-access contents

bb) sending from the server to the client of an authentication module to be run in the client

cc) execution of an authentication protocol for authenticating the mobile security module and, where

appropriate, its holder by means of the authentication module

dd) if the authentication in step cc) was successful, addition to the request in step aa) of a session ID which was generated in the course of the communications between the authentication module and the server

ee) sending of the new request to the server application

ff) checking of the session ID in the request to see that it is recorded in the server

gg) processing of the content requested for transmission and searching of the content for further links to other protected-access contents

hh) addition of the session ID to the links identified

ii) sending of the content modified as in step hh) to the client.

2. Method according to claim 1, characterized in that the server is a web server and the protected contents are web pages which are called up via a browser by a URL request from a client.

3. Method according to claim 1, characterized in that the authentication protocol is executed in the followed steps:

jj) generation of a random number by the server application when the content requested is access-protected and the requirements for access have not been satisfied, and sending of the random number to the authentication module

kk) sending of the random number from the authentication module to the mobile security module

ll) generation in the mobile security module of a digital signature which takes account of the identity number of the mobile security module, the random number and the key of the mobile security module

mm) sending of the digital signature to the server

nn) checking of the correctness of the digital signature using the security module of the server.

4. Method according to claim 2, characterized in that the server application is a servlet and the client authentication module is an authentication applet and in that on receipt of a URL

request the servlet checks the URL request for the presence of a session ID and if there is no session ID present sends an authentication applet containing a random number to the client.

5. Method according to claim 1, characterized in that the communications between client and server take place via SSL (secure sockets layer) as the transmission protocol.

6. Method according to claim 4, characterized in that the authentication applet communicates with the servlet by internet or intranet using the TCP/IP protocol.

7. Method according to claim 3, characterized in that the digital signature is generated by means of a symmetrical encryption algorithm with the help of a secret key agreed between client and server, or by means of an asymmetrical encryption algorithm with the help of a private key, the server being in possession of the public key.

8. Method according to claim 7, characterized in that the symmetrical encryption algorithm is DES or triple DES and the asymmetrical encryption algorithm is RSA, DSA or an elliptic curve algorithm.

9. Method according to claim 4, characterized in that if the digital signature does not agree, the servlet sends an error message to the client applet.

10. Method according to claim 1, characterized in that a session ID is generated from a large range of values to prevents its being discovered by a targeted search.

11. Method according to claim 1, characterized in that the session ID shows the user to be an authorized person for all requests within a specified session.

12. Method according to claim 1, characterized in that the session ID is given a period of validity.

13. Method according to claim 12, characterized in that the session ID loses its validity on expiry of a fixed time or when a session is terminated by means of a log-off page.

14. Method according to claim 1, characterized in that the session ID generated in step dd) is recorded in a table and in that the presence of an entry in the table is a requirement for access to all the protected-access pages.

15. Method according to claim 14, characterized in that when the validity of a session ID expires or when a session is terminated by means of a log-off page the session ID is deleted from the table.

16. Apparatus comprising at least the following components:

a) client comprising at least:
aa) a browser
bb) a computer software product for executing steps aa), cc), dd) and ee) of the method according to claim 1
cc) reader for a mobile security module

b) server comprising at least:
aa) a computer software product for executing steps bb), ff), gg), hh) and ii) of the method according to claim 1

c) a communications connection between client and server.

17. Apparatus according to claim 16, characterized in that the server is a web server and in that the communications connection between client and web server is made by internet or intranet.

18. Web server comprising at least:

a) a non-volatile memory for storing web pages

b) a computer software product for executing steps bb), ff), gg), hh) and ii) of the method according to claim 1

19. Web server according to claim 18, characterized in that a security module for performing step nn) of the method according to claim 1 is also provided.

20. Client comprising at least:

aa) a browser

bb) a computer software product for executing steps aa), cc), dd) and ee) of the method according to claim 1.

21. Client according to claim 20, also comprising:

a) chip card reader for a mobile security module

b chip card having a non-volatile, protected memory containing at least:

aa) a card number

bb) a cryptographic key.

22. Computer software product which is stored in the internal memory of a digital computer, comprising items of software

code for executing the method according to claims 1-15 when the product is run on the computer.